

Guía paso a paso: Proyecto Maven en Visual Studio Code e IntelliJ IDEA

Esta guía explica cómo crear, desde cero y directamente en cada IDE, un proyecto Maven sencillo en Java que convierte un archivo XML a JSON. Se cubren dos entornos: Visual Studio Code e IntelliJ IDEA.

1. Crear el proyecto Maven en Visual Studio Code

1.1. Preparar Visual Studio Code

1. Abre Visual Studio Code.
2. Ve a la pestaña de Extensiones.
3. Instala las siguientes extensiones:
 - Extension Pack for Java
 - Maven for Java (si no viene incluida).

1.2. Crear un nuevo proyecto Maven

1. Pulsa Ctrl + Shift + P para abrir la Command Palette.
2. Escribe “Maven: Create Maven Project” y selecciona la opción.
3. Selecciona el archetype: maven-archetype-quickstart.
4. Acepta la versión por defecto del archetype.
5. Introduce los datos del proyecto:
 - Group Id: com.ejemplo
 - Artifact Id: xml-json-ejercicio
6. Elige la carpeta donde se creará el proyecto y confirma sus propiedades en la consola.
7. Cuando VS Code te pregunte si quieres abrir el proyecto, selecciona abrir (Open).

Después de esto, tendrás una estructura de proyecto similar a:

xml-json-ejercicio/
pom.xml
src/main/java/...

8. Si da error JAVA_HOME, sigue los siguientes pasos:

- Abrir el Símbolo del sistema:
 - Pulsar Win + R, escribir cmd y pulsar Enter.
- Escribir el siguiente comando y pulsar Enter:
`java -version`
- Si Java está instalado y en el PATH, debería aparecer algo como:
`java version "21.0.x" ...`
`openjdk version "21.0.x" ...`
- Si aparece un mensaje del tipo "'java' no se reconoce como un comando interno o externo", significa que Java no está instalado correctamente o no está en el PATH.
- Buscar en el menú inicio "Variables de entorno".
- Abrir "Editar las variables de entorno del sistema".
- Pulsar el botón "Variables de entorno...".
- En "Variables del sistema" (o en "Variables de usuario", según se prefiera), pulsar "Nueva...".
- Rellenar:
 - Nombre de la variable: JAVA_HOME
 - Valor de la variable: C:\Program Files\Java\jdk-21 (Ajustar la ruta exacta al directorio donde esté instalado el JDK en cada equipo.)
- En la misma ventana de Variables de entorno, buscar la variable Path en "Variables del sistema".
- Seleccionarla y pulsar "Editar".
- Pulsar "Nuevo" y escribir:
`%JAVA_HOME%\bin`
- Aceptar todas las ventanas para guardar los cambios.
- Cerrar las consolas y Visual Studio Code que estuvieran abiertos y abrir de nuevo.

1.3. Editar pom.xml en VS Code

1. Ver la versión de Java desde Visual Studio Code

- Ve al panel de Comandos (Ctrl + Shift + P)
 - Escribe: Java: Configure Java Runtime y pulsa Enter.
 - Se abrirá una ventana con los JDKs detectados y cuál está usando VS Code para los proyectos.
2. En el panel Explorer, haz clic en pom.xml para abrirlo.

3. Añade la sección de properties (si no existe) para fijar la versión de Java:

```
<properties>
    <maven.compiler.source>tu versión de java</maven.compiler.source>
    <maven.compiler.target>tu versión de java</maven.compiler.target>
</properties>
```

3. Dentro de <dependencies> añade las dependencias de Jackson para XML y JSON:

```
<dependencies>
    <!-- Jackson core / JSON -->
    <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
        <version>2.16.1</version>
    </dependency>

    <!-- Jackson XML -->
    <dependency>
        <groupId>com.fasterxml.jackson.dataformat</groupId>
        <artifactId>jackson-dataformat-xml</artifactId>
        <version>2.16.1</version>
    </dependency>
</dependencies>
```

VS Code descargará automáticamente las dependencias (puede tardar un poco, pero no necesitas usar la terminal).

1.4. Crear el archivo XML de ejemplo (yo.xml)

1. En el panel Explorer de VS Code:

- Haz clic derecho sobre src/main y selecciona New Folder. Llámala resources.

2. Haz clic derecho sobre la carpeta resources y selecciona New File. Nómbralo yo.xml.

3. Pega el siguiente contenido en yo.xml y edítalo con tus datos:

```
<?xml version="1.0" encoding="UTF-8"?>

<yo>
    <nombre>Tu nombre</nombre>
    <edad>Tu edad</edad>
</yo>
```

1.5. Crear la clase Java XmlToJson

1. En src/main/java/com/ejemplo, crea un nuevo archivo Java llamado XmlToJson.java.

2. Pega el siguiente código:

```
package com.ejemplo;

import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;
import com.fasterxml.jackson.dataformat.xml.XmlMapper;

import java.io.IOException;
import java.io.InputStream;
import java.nio.charset.StandardCharsets;

public class XmlToJson {

    public static void main(String[] args) {
        try {
            // Cargar el XML desde resources/yo.xml
            InputStream is = XmlToJson.class.getClassLoader().getResourceAsStream("yo.xml");
            if (is == null) {
                System.out.println("No se encontró yo.xml en resources");
                return;
            }

            String xml = new String(is.readAllBytes(), StandardCharsets.UTF_8);

            // Mapper para XML -> JsonNode
            XmlMapper xmlMapper = new XmlMapper();
            JsonNode node = xmlMapper.readTree(xml.getBytes(StandardCharsets.UTF_8));

            // Mapper para JsonNode -> String JSON
            ObjectMapper jsonMapper = new ObjectMapper();

            String json = jsonMapper.writerWithDefaultPrettyPrinter()
                .writeValueAsString(node);

            System.out.println("XML original:\n" + xml);
            System.out.println("\nJSON resultante:\n" + json);

        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

1.6. Ejecutar el programa en VS Code

1. Abre el archivo XmlToJson.java.
2. Haz clic en Run para ejecutar el programa.
3. En la salida deberías ver el XML original y el JSON generado a partir de yo.xml.
4. Si te da problemas ejecuta mvn -v en Powershell
5. Si aparece: "mvn no se reconoce como un comando interno o externo", entonces Maven no está instalado o no está en el PATH.

1.7. Instalación de Maven (solo si aparece error de 1.6.5)

1. Ve a la página oficial de descargas de Apache Maven (búscalas en Google como "apache maven download").
2. Descarga el archivo ZIP binario, por ejemplo: apache-maven-3.x.x-bin.zip.
3. Haz clic derecho sobre él y elige "Extraer todo...".
4. Elige una carpeta sin espacios en la ruta, por ejemplo: C:\apache-maven-3.x.x\ o C:\dev\apache-maven-3.x.x\.
5. Pulsa en el botón Inicio de Windows y escribe: Editar las variables de entorno del sistema
6. En la ventana Propiedades del sistema, haz clic en el botón "Variables de entorno...".
7. En la parte inferior (Variables del sistema), busca la variable Path.
8. Selecciona Path y pulsa "Editar...".
9. Pulsa "Nuevo" y añade la ruta a la carpeta bin de Maven, por ejemplo: C:\apache-maven-3.9.9\bin (ajusta la versión a la que hayas descargado).
10. Acepta todas las ventanas pulsando en "Aceptar".
11. Ejecuta mvn clean compile en Powershell
12. Vuelve al paso 1.6.1.

2. Crear el proyecto Maven en IntelliJ IDEA

2.1. Crear un nuevo proyecto Maven

1. Abre IntelliJ IDEA.

2. En la pantalla de bienvenida, selecciona New Project.

3. En la lista de la izquierda, elige Maven.

4. Configura:

- Language: Java
- Build system: Maven
- JDK: una versión instalada

5. Pulsa Next o Create

6. Introduce los datos del proyecto:

- GroupId: com.ejemplo
- ArtifactId: xml-json-ejercicio

7. Elige la carpeta de destino y pulsa Finish.

2.2. Configurar pom.xml en IntelliJ

1. Ver la versión de Java desde IntelliJ

- File > Project Structure (Ctrl + Alt + Shift + S)
- En la sección Project / Project SDK → versión JDK

2. En el panel Project,, haz clic en pom.xml para abrirlo.

3. Añade la sección de properties (si no existe) para fijar la versión de Java:

```
<properties>
    <maven.compiler.source>tu versión de java</maven.compiler.source>
    <maven.compiler.target>tu versión de java</maven.compiler.target>
</properties>
```

4. Dentro de <dependencies>, añade las dependencias de Jackson:

```
<dependencies>
    <!-- Jackson core / JSON -->
    <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
        <version>2.16.1</version>
    </dependency>

    <!-- Jackson XML -->
    <dependency>
        <groupId>com.fasterxml.jackson.dataformat</groupId>
        <artifactId>jackson-dataformat-xml</artifactId>
        <version>2.16.1</version>
    </dependency>
</dependencies>
```

4. Guarda el archivo. Si IntelliJ no recarga Maven automáticamente, haz clic derecho en pom.xml y selecciona Maven > Reload project.

2.3. Crear el archivo yo.xml en resources

1. En el panel Project:

- Haz clic derecho en src/main y elige New > Directory. Nómbralo resources.

2. Haz clic derecho en la carpeta resources y selecciona New > File. Nómbralo yo.xml.

3. Pega el siguiente contenido en yo.xml y edítalo con tus datos:

```
<?xml version="1.0" encoding="UTF-8"?>
<yo>
    <nombre>Tu nombre</nombre>
    <edad>Tu edad</edad>
</yo>
```

2.4. Crear la clase XmlToJson en IntelliJ

1. En src/main/java, crea un paquete llamado com.ejemplo (New > Package).

2. Haz clic derecho sobre el paquete com.ejemplo y elige New > Java Class.

3. Ponle de nombre XmlToJson.

4. Pega el mismo código que usamos en Visual Studio Code:

```
package com.ejemplo;

import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;
import com.fasterxml.jackson.dataformat.xml.XmlMapper;

import java.io.IOException;
import java.io.InputStream;
import java.nio.charset.StandardCharsets;

public class XmlToJson {

    public static void main(String[] args) {
        try {
            // Cargar el XML desde resources/yo.xml
            InputStream is = XmlToJson.class.getClassLoader().getResourceAsStream("yo.xml");
            if (is == null) {
                System.out.println("No se encontró yo.xml en resources");
                return;
            }

            String xml = new String(is.readAllBytes(), StandardCharsets.UTF_8);

            // Mapper para XML -> JsonNode
            XmlMapper xmlMapper = new XmlMapper();
            JsonNode node = xmlMapper.readTree(xml.getBytes(StandardCharsets.UTF_8));

            // Mapper para JsonNode -> String JSON
            ObjectMapper jsonMapper = new ObjectMapper();
            String json = jsonMapper.writerWithDefaultPrettyPrinter()
                .writeValueAsString(node);

            System.out.println("XML original:\n" + xml);
            System.out.println("\nJSON resultante:\n" + json);

        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

2.5. Ejecutar el programa en IntelliJ

1. Abre la clase XmlToJson en el editor.
2. Selecciona Run 'XmlToJson.main()'



Centro Oficial FP
Digital & Tech

3. En la ventana Run, verás el XML original y el JSON generado.